

Oracle 8i Rel. 3 (8.1.7) unter Redhat Linux 8.0 installieren

Vorbemerkungen

Diese Anleitung basiert im wesentlichen auf dem Howto "INSTALLING ORACLE 8I ON REDHAT LINUX 7 OR 8" von "Bobo's Log - Cries for help", zu erreichen über die URL:

<http://seneca.omniscia.org/~vivake/log/archives/000041.html>. Vielen Dank an dieser Stelle an den Bobo dass dieses Tutorial existiert, ansonsten wäre eine erfolgreiche Installation bei mir wahrscheinlich mit sehr(!) viel mehr Aufwand verbunden gewesen.

Ich habe u.a. einige Anpassungen und Erweiterungen vorgenommen, die bei meiner Installation erforderlich waren, um erfolgreich eine Datenbank aufsetzen zu können.

Anmerkungen, Kritik (positive wie negative, hauptsache konstruktiv) und Hinweise zu Fehlern nehme ich gerne unter stefan@alles-eazy.de entgegen.

Nötige Voraussetzungen zur Installation

Zunächst benötigen Sie natürlich die *Installations-CD von Oracle 8.1.7.0 für Linux*. Zusätzlich sollten Sie das *Patchset auf 8.1.7.4* zur Hand haben.

Da Oracle die Glibc-Version 2.1.3 voraussetzt, bei Redhat 8.0 aber die Version 2.2 installiert ist, müssen zunächst die entsprechenden Libraries auf diesen Stand gebracht werden.

Laden Sie vom Redhat-FTP-Server (oder einem Mirror) aus dem Verzeichnis:

pub/redhat/linux/7.3/de/os/i386/RedHat/RPMS/

die Dateien:

- *compat-egcs-6.2-1.1.2.16.i386.rpm*
- *compat-glibc-6.2-2.1.3.2.i386.rpm*
- *compt-libs-6.2-3.i386.rpm*
- *compat-libstdc++-6.2-2.9.0.16.i386.rpm*

herunter. Ausserdem benötigen Sie die Datei *glibc-2.1.3-stubs.tar.gz*. Diese ist z.B. nach einer kostenlosen Registrierung über technet.oracle.com herunterladbar oder auf dem SuSE-FTP-Server im Verzeichnis:

ftp://ftp.suse.com/pub/suse/discontinued/i386/commercial/Oracle/

zu finden (als User "ftp" mit der eMail-Adresse als Passwort anmelden!)

Zusätzlich installieren wir noch das aktuelle Java-Runtime-Environment (JRE). Dieses finden Sie unter der URL:

<http://java.sun.com/j2se/1.4.1/download.html>

Wählen Sie die das JRE der Datei "Linux RPM in self-extracting file".

Anmerkung: Um den Oracle-Installer aufrufen zu können, benötigen Sie eine funktionierende X-Window-Installation (z.B. Gnome oder KDE).

Erste Vorbereitungen

Zuerst müssen die compat-Libraries als root installiert werden:

```
> rpm --install compat-libs-6.2-3.i386.rpm
> rpm --install compat-libstdc++-6.2-2.9.0.16.i386.rpm
> rpm --install compat-glibc-6.2-2.1.3.2.i386.rpm
> rpm --install compat-egcs-6.2-1.1.2.16.i386.rpm
```

Sollten noch Pakete installiert sein, die mit den o.g. kollidieren, entfernen sie diese bitte mit

```
> rpm --nodeps -e [paketname].
```

Installieren Sie nun das JRE:

```
> chmod 755 j2re-1_4_1_01-linux-i586-rpm.bin
> ./j2re-1_4_1_01-linux-i586-rpm.bin
> rpm --install j2re-1_4_1_01-linux-i586.rpm
```

Damit die Installation von Oracle ordnungsgemäß durchläuft, sollte die maximale Größe des Shared-Memory (shmmax) erhöht werden. Die voreingestellte Größe sind 32MB. Oracle schlägt jedoch vor, hier die Hälfte des Arbeitsspeichers als Maximalgröße einzutragen. Dies machen Sie mit:

```
> echo [halber arbeitsspeicher in byte] > /proc/sys/kernel/shmmax
```

bei 512MB RAM also:

```
> echo 268435456 > /proc/sys/kernel/shmmax
```

Damit nach einem System-Neustart der Wert von SHMMAX wieder auf dieser Größe steht, können Sie SHMMAX ja z.B. im Startup-Script von Oracle neu setzen (Siehe Beispielstartscript in Anhang B). Alternativ lässt sich der Wert auch über die Datei /etc/sysctl.conf setzen. Hier muss als root nur der Eintrag:

```
kernel.shmmax = 268435456
```

eingetragen werden.

Zuletzt muss noch der Befehl /bin/df angepasst werden, damit der Datenbank-Assistent dbassist bei der Überprüfung auf freien Speicherplatz nicht hängenbleibt:

```
> mv /bin/df /bin/df.orig
> cat << EOF > /bin/df
#!/bin/sh
/bin/df.orig -P \${@}
EOF
> chmod 755 /bin/df
```

Benutzer und Verzeichnisse anlegen

Da der Oracleserver nicht als root installiert werden darf, legen wir einen Benutzer namens "oracle" an, der diese Aufgaben übernehmen soll. Dieser muss den Gruppen dba, oracle und oinstall (als primäre Gruppe) zugewiesen werden:

```
> groupadd dba
> groupadd oracle
> groupadd oinstall
> useradd oracle -g oinstall -G dba,oracle
```

Melden Sie sich nun auf einer anderen Console kurz als Benutzer oracle an und wieder ab, damit das Home-Verzeichnis mit den zugehörigen Profil-Dateien angelegt wird.

Danach arbeiten Sie als root weiter:

- Ergänzen Sie die Datei /home/oracle/.bash_profile um folgende Zeilen:

```
# user specific environment and startup programs
. /usr/i386-glibc21-linux/bin/i386-glibc21-linux-env.sh
export LD_ASSUME_KERNEL=2.2.5

umask 022

export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=$ORACLE_BASE/product/8.1.7

export
PATH=$ORACLE_HOME/bin:$ORACLE_HOME/ctx/lib:/usr/bin:/bin:/usr/local/bin:$PATH
export
LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib:/lib:/usr/local/lib:$LD_LIBRARY_PATH
export ORA_NLS33=$ORACLE_HOME/ocommon/nls/admin/data
export ORACLE_SID=TEST
```

(Eine vollständige Datei .bash_profile finden Sie in Anhang A - .bash_profile)

Verzeichnisstrukturen für Oracle anlegen

Legen Sie ggfs. noch nicht existierende Verzeichnisse für Oracle an. Dazu sollten Sie als Benutzer oracle angemeldet sein:

```
> mkdir /u01
> mkdir $ORACLE_BASE
> mkdir $ORACLE_BASE/product
> mkdir $ORACLE_HOME
> mkdir $ORACLE_HOME/bin
> cd $ORACLE_HOME/bin
> ln -s /usr/bin/i386-glibc21-linux-gcc cc
> ln -s /usr/bin/i386-glibc21-linux-gcc gcc
```

Oracle installieren

(Um den Oracle-Installer aufrufen zu können, müssen Sie sich in X-Window befinden!)

Wechseln Sie jetzt in das Verzeichnis mit Ihren Oracle-Installationsdateien (z.B. /mnt/cdrom/)

Bei einigen Rechnern lässt sich der Oracle-Installer nicht starten, wenn die LANG-Umgebungsvariable gesetzt ist. Deswegen leeren wir diese im Vorfeld mit:

```
> export LANG=
```

Jetzt starten wir die Installation:

```
> ./runInstaller
```

Wenn Sie nach dem Verzeichnis des JRE gefragt werden (vorbelegt mit `/usr/local/java`), ändern Sie dies auf:

`/usr/java`

Wählen Sie die Installations-Einstellungen nach Ihren Wünschen und starten Sie die Installation. Bei der Installation treten bei mir jedes Mal zwei Fehler auf: Zum einen kann (angeblich) eine Library nicht installiert werden, zum zweiten tritt kurz vor dem Ende der Installation ein Make-Fehler auf. Diese können Sie mit gutem Gewissen ignorieren, die Library wird trotz des Fehlers ordnungsgemäß in das richtige Verzeichnis installiert, der Make-Fehler wird durch die Installation der GLIBC-Stubs (s.u.) wieder "repariert".

Zum Ende der Installation werden Sie aufgefordert, das Script `$ORACLE_HOME/root.sh` als root auszuführen. Bevor wir dies tun, müssen wir in dem Script noch einige Fehler berichtigen:

Öffnen Sie ein Consolen-Fenster und melden Sie sich als root an:

```
> su -
```

Ändern Sie nun die Datei `$ORACLE_HOME/root.sh` wie folgt:

Zeile 98

```
SED=/usr/local/bin/sed
in:
SED=/bin/sed
```

Zeile 102

```
RMF=/bin/rm -f
in:
RMF="/bin/rm -f "
```

Zeile 156

```
RUID=`/usr/bin/id|AWK -F\ ( '{print $2}'|AWK -F\ ) '{print $1}`
in:
RUID=`/usr/bin/id|AWK -F\ ( '{print $2}'|AWK -F\ ) '{print $1}'`
```

Nun können Sie das Script starten:

```
> cd $ORACLE_HOME
> ./root.sh
```

Mit `exit` verlassen Sie die Root-Shell wieder.

Bevor wir jetzt mit der Installation fortfahren, installieren wir als User oracle über ein Consolen-Fenster die GLIBC-Stubs:

- Kopieren Sie die Datei *glibc-2.1.3-stubs.tar.gz* in das Verzeichnis `$ORACLE_HOME`
- Installieren Sie diese dann wie folgt:

```
> cd $ORACLE_HOME
> tar xzvf glibc-2.1.3-stubs.tar.gz
> ./setup_stubs.sh
> cd bin
> relink all
```

Nachdem der Relink durchgelaufen ist, können Sie die Installation des Oracle-Installers bis zum Ende durchlaufen lassen.

Installation der Patches

Bevor wir den Installer nun schliessen, installieren wir noch die Oracle-Patches für die Version 8.1.7.4. Im Beispiel liegen diese im Verzeichnis `/home/oracle/oracle_install/patches` entpackt vor.

- Klicken Sie also auf "Weitere Pakete installieren"
- Wählen Sie als Quellmedium das Verzeichnis `/home/oracle/oracle_install/patches/stage/products.jar` aus.
- Starten Sie nun die Installation, alle installierten Pakete werden nun aktualisiert.

Tritt bei der Installation der Fehler auf, dass die Datei „oracle“ nicht installiert werden kann, stoppen Sie als Benutzer oracle von einem Consolen-Fenster aus die Datenbank mit den Befehlen: `dbshut` und `lsnrctl stop`. Danach können Sie über den Button "Wiederholen" normal mit der Installation fortfahren. Nach dem Update können Sie dann Oracle mit `dbstart` und `lsnrctl start` wieder hochfahren.

Einrichten des Startup-Scripts

Um Oracle nun nach jedem Neustart des Systems automatisch aktivieren zu können, benötigen wir noch ein Startup-Script (siehe Anhang B - Oracle-Startupscript) und einen Eintrag der zu startenden Oracle-Instanzen in der Datei `/etc/oratab`.

Heisst die eingerichtete Oracle-Instanz (SID) z.B. TEST, dann lautet der benötigte Eintrag in der Datei `/etc/oratab` so:

```
TEST:/u01/app/oracle/product/8.1.7:Y
```

Das Startup-Script erstellen wir in einer Root-Console unter dem Namen "oracle" im Verzeichnis `/etc/init.d`. Danach können Sie es auf folgendem Wege aktivieren:

```
> chown root:root /etc/init.d/oracle
> chmod 755 /etc/init.d/oracle
> chkconfig --add oracle
```

Sie können nun Oracle einfach mit dem Befehl:

```
> service oracle start
```

hochfahren und mit

```
> service oracle stop
```

herunterfahren.

Fertig ;)

Die Oracle-Installation ist nun abgeschlossen. Mit dem Assistenten `dbassist` können Sie als Benutzer oracle nun neue Datenbanken anlegen, bestehende Datenbanken löschen oder administrieren. Mit `netca` ist es möglich, den Listener einzurichten.

Anhang

Anhang A - .bash_profile

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs
. /usr/i386-glibc21-linux/bin/i386-glibc21-linux-env.sh
export LD_ASSUME_KERNEL=2.2.5

umask 022

export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=$ORACLE_BASE/product/8.1.7

export
PATH=$ORACLE_HOME/bin:$ORACLE_HOME/ctx/lib:/usr/bin:/bin:/usr/local/bin:$PATH
export
LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib:/lib:/usr/local/lib:$LD_LIBRARY_PATH
export ORA_NLS33=$ORACLE_HOME/ocommon/nls/admin/data
export ORACLE_SID=TEST

PATH=$PATH:$HOME/bin

export PATH
unset USERNAME
```

Anhang B - Oracle-Startupscript

```
#!/bin/sh
#
# Startup script for Oracle
#
# chkconfig: 345 79 11
# description: Oracle.
# processname: oracle
# pidfile: /var/run/oracle.pid
#
# Source function library.
. /etc/rc.d/init.d/functions

ORACLE_HOME=/u01/app/oracle/product/8.1.7
ORACLE_USER=oracle

# See how we were called
case "$1" in
    start)
        echo -n "Starting Oracle Database Server:"
        daemon --user=$ORACLE_USER $ORACLE_HOME/bin/dbstart
        success "Starting Oracle Database Server:"
        echo
        echo -n "Starting Oracle TNS Listener:"
        daemon --user=$ORACLE_USER $ORACLE_HOME/bin/lsnrctl start >>/dev/null
        success "Starting Oracle TNS Listener:"
        echo
        touch /var/lock/subsys/oracle
        ;;
    stop)
        echo -n "Stopping Oracle Database Server:"
        daemon --user=$ORACLE_USER $ORACLE_HOME/bin/dbshut
        success "Stopping Oracle Database Server:"
        echo
        echo -n "Stopping Oracle TNS Listener:"
        daemon --user=$ORACLE_USER $ORACLE_HOME/bin/lsnrctl stop >>/dev/null
        success "Stopping Oracle TNS Listener:"
        echo
        rm -f /var/lock/subsys/oracle
        rm -f /var/run/oracle.pid
        ;;
    restart)
        $0 stop
        sleep
        $0 start
        ;;
    *)
        echo "Usage: $0 {start|stop|restart}"
        exit 1
esac

exit 0
```